

## PENENTUAN RUTE TERPENDEK TUJUAN WISATA DI KOTA TOBOALI MENGUNAKAN ALGORITME DIJKSTRA BERBASIS WEB

Fransiskus Panca Juniawan<sup>\*1</sup>, Dwi Yuny Sylfania<sup>2</sup>

<sup>1,2</sup>Program Studi Teknik Informatika, STMIK Atma Luhur  
Email: <sup>1</sup>fransiskus.pj@atmaluhur.ac.id, <sup>2</sup>dysylfania@atmaluhur.ac.id

<sup>\*</sup>Penulis Korespondensi

(Naskah masuk: 22 April 2019, diterima untuk diterbitkan: 14 Januari 2020)

### Abstrak

Kota Toboali merupakan ibukota Kabupaten Bangka Selatan yang telah menjadi daerah tujuan wisata baru karena keindahan yang dimiliki dan didukung oleh kebijakan pemerintah daerah. Oleh sebab itu fasilitas wisata terus dikembangkan hingga saat ini. Namun banyak wisatawan dari daerah lain yang mengalami kesulitan dalam perjalanannya menuju tujuan wisata di Kota Toboali. Hal ini dikarenakan mereka belum mengenal letak geografis Kota Toboali. Kesulitan yang dialami para wisatawan ini adalah dalam hal mengetahui dan menentukan jalan yang akan dilalui menuju ke tujuan wisatanya. Untuk itu dibuatlah sistem informasi geografis yang berbasis *web* dan dilengkapi dengan fungsi penentuan jarak terpendek menggunakan Algoritme Dijkstra agar dapat lebih cepat sampai pada tujuan wisata. Sistem dirancang berbasis *web* agar lebih banyak orang yang menggunakannya. Dalam penelitian ini juga dibahas mengenai cara perhitungan manual algoritme Dijkstra, selain itu untuk pembuktian hasil perhitungan, diberikan pula berupa contoh kasus penentuan jarak terpendek di salah satu sudut Kota Toboali. Hasil dari penelitian berupa sistem informasi pariwisata Kota Toboali yang dapat menentukan jarak terpendek menuju lokasi wisata. Dari hasil pengujian algoritme dapat dibuktikan bahwa sistem mampu menentukan jarak terpendek dari titik awal yang ditentukan pengguna menuju titik tujuan wisatanya. Dari pengujian *blackbox* didapat hasil bahwa fungsional sistem memiliki kinerja yang baik.

**Kata kunci:** Sistem Informasi Geografis, Dijkstra, Rute Terpendek, Pariwisata Kota Toboali

## SHORTEST PATH DETERMINATION OF THE TOURIST DESTINATION IN TOBOALI USING WEB-BASED DIJKSTRA ALGORITHM

### Abstract

Toboali City is the capital of South Bangka Regency which has become a new tourist destination because of its beauty and supported by local government policies. Therefore tourism facilities continue to be developed to date. However, many tourists from other regions experience difficulties on their way to tourist destinations in Toboali City. This is because they do not know the geographical location of Toboali City. The difficulty experienced by these tourists is in terms of knowing and determining the path to be traveled to the destination. For this reason, a geographic information system based on the web was made and equipped with the shortest distance determination function using the Dijkstra algorithm to be able to reach tourist destinations faster. The system is designed web-based so that more people use it. In this study also discussed how the manual calculation of the Dijkstra algorithm, in addition to proof of calculations, is also given in the form of a case in determining the shortest distance in one corner of the city of Toboali. The results of the research in the form of the Toboali City tourism information system that can determine the shortest distance to the tourist location. From the algorithm test results it can be proven that the system is able to determine the shortest distance from the user-specified starting point to the destination point of the tour. From the blackbox testing, the results show that functional systems have good performance.

**Keywords:** Geographic Information System, Dijkstra, Shortest Path, Toboali City Tourism

### 1. PENDAHULUAN

Kota Toboali merupakan ibukota Kabupaten Bangka Selatan yang memiliki beberapa tujuan

wisata di dalam kota dan sekitarnya. Diantaranya adalah Pantai Batu Kodok, Pantai Batu Perahu, Pantai Gunung Namak, Pantai Tanjung Labun,

Pantai Batu Ampar, Pantai Payak Ubi, Pantai Pulau Dapur dan Pantai Tanjung Kelayang (Rakyat Pos, 2017). Kota ini juga mengikuti kebijakan Gubernur Bangka Belitung yang memiliki visi untuk menjadikan bidang pariwisata sebagai salah satu fokus pembangunan di provinsi ini (Perda Basel, 2016). Selain itu, sector pariwisata juga menjadi salah satu dari 11 (sebelas) program unggulan Bangka Belitung, dimana pariwisata merupakan salah satu program yang menjadi prioritas utama (Nonadp, 2017). Daerah-daerah yang memiliki potensi wisata telah dipercantik dan dikembangkan untuk menarik minat wisatawan domestik maupun mancanegara.

Sebagai nama yang masih baru pada kancah pariwisata di Indonesia, banyak wisatawan dari daerah lain yang masih mengalami kesulitan untuk mencapai tujuan wisata di Kabupaten Bangka Selatan, terutama di Kota Toboali. Untuk itu dibutuhkan sebuah sistem informasi berbasis geografis yang dapat menyediakan data seluruh lokasi wisata di Kabupaten Bangka Selatan pada umumnya dan Kota Toboali pada khususnya. Selain itu, permasalahan lainnya adalah untuk mencapai tujuan wisata di Kota Toboali kita harus melewati banyak jalan yang bercabang. Hal tersebut menjadi permasalahan tersendiri bagi wisatawan non-lokal dan wisatawan asing. Apabila mereka salah mengikuti jalur atau melalui jalur yang memutar, maka tentu saja mereka akan dirugikan dalam hal waktu dan tenaga.

Untuk mengatasi permasalahan tersebut maka diusulkan penerapan fungsi penentuan jarak terpendek pada tujuan-tujuan wisata di Kota Toboali agar dapat memberikan informasi berupa jalur terpendek yang dapat dilalui untuk mencapai tujuan wisata.

Ada banyak metode yang dapat digunakan untuk penentuan jarak terpendek, salah satunya adalah Algoritme Dijkstra (Wang, Zhang, & Cui, 2017). Algoritme ini dipilih karena merupakan algoritme yang efisien dan cepat dalam penentuan rute terpendek (Ojekudo & Akpan, 2017). Selain itu algoritme ini mudah digunakan dengan *node-node* sederhana pada jaringan yang tidak rumit (Risald, Mirino, & Suyoto, 2017).

Telah dilakukan penelitian terkait algoritme Dijkstra yang pernah dilakukan seperti pada penentuan jarak terpendek menggunakan Dijkstra (Ojekudo & Akpan, 2017), penentuan hotel terdekat menggunakan Dijkstra (Miee, Ahmed, & Ahmed, 2017). Dengan konsumsi oli pada kendaraan, dijkstra dapat digunakan untuk penentuan rute kendaraan di perkotaan (Jin-dong Zhang et al., 2016). Ada juga penerapannya di sistem informasi geografis pelayanan kesehatan (Raja, N, & Irwansyah, 2015), pencarian lokasi agen bus dan travel (Kurniawan, Nurhayati, & Martono, 2015), pencarian pendonor darah terdekat (Hossain, Das, Patwary, & Hassan, 2018), dan pencarian lapangan

futsal (Wahyuningsih & Syahreza, 2018). Dibuat juga aplikasi untuk menampilkan hasil perhitungan dan visualisasi jarak terpendek dalam proses pengantaran barang (Kusuma & Agung, 2019).

Penelitian lainnya melakukan perbandingan kinerja algoritme penentuan jarak terpendek antara dijkstra dan Floyd-warshall (Risald et al., 2017). Variasi penggabungan algoritme juga dibuat untuk menentukan jarak terpendek dengan dijkstra dan algoritme *divide and conquer* (Hou & Zhang, 2017).

Modifikasi dan pengembangan algoritme dijkstra untuk perencanaan rute transportasi publik (Bozyigit, Alankus, & Nasiboglu, 2017), Dijkstra juga dapat digunakan pada penentuan tempat parkir kosong terdekat (Wang et al., 2017) dan juga sebagai petunjuk untuk pintu keluar terdekat dari tempat parkir (Wu, Wu, & Song, 2017). Selanjutnya ada penentuan rute terbaik *e-parking* berbasis GIS (Prianto & Kusnadi, 2018). Diusulkan juga metode baru dengan penggabungan algoritme dijkstra dengan Bellman-Ford untuk penentuan jarak terpendek berbasis *parallel vertex* (Vesovi, Smiljani, & Kostic, 2016). Dikembangkan juga navigasi dalam ruangan, rute terpendeknya, dan penggunaan *QR Code* untuk penentuan posisi dalam ruangan (Ginardi & Munif, 2016). Pada gedung bertingkat sistem navigasi *indoor* juga dibuat menggunakan *bidirectional dijkstra search* (Ardhiansyah, Putra, Ginardi, & Munif, 2016).

Pada *path planning* juga dapat diterapkan Dijkstra untuk penentuan jarak terpendek. Ada kendaraan pemandu otomatis (Qing, Zheng, & Yue, 2017), pesawat pengangkut berbasis geometri (Jing Zhang, Yu, Qu, & Wu, 2017), distribusi optimal menggunakan kolaborasi dijkstra dengan algoritme *sweep* (Zulfiqar, Isnanto, & Nurhayati, 2018), dan sistem transportasi cerdas berbasis prediksi *short-term traffic flow* (Zhu, Du, Sun, & Cao, 2018).

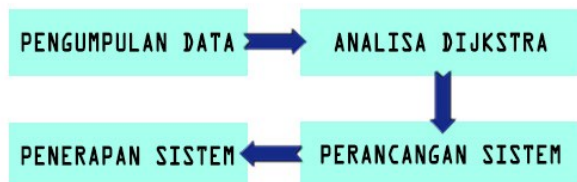
Dijkstra juga dapat diterapkan bukan hanya pada penentuan jarak terpendek. Dijkstra dapat diterapkan sebagai alat bantu akselerasi pada *processor* berbasis *multi-core* (Prasad, Krishnamurthy, & Kim, 2018). Diterapkan juga pada *power system* berdasar *sequential restoration strategy* (Goo, Jung, & Hur, 2016), pendekatan penglihatan *stereo* untuk pendeteksian jalan dengan *vanishing point* (Y. Zhang, Su, Yang, Ponce, & Kong, 2018). Permasalahan komposisi *web service* juga dapat diselesaikan dengan penerapan dijkstra (Moo, Hernández, Uc, & Madera, 2016) dan pada *internet of things* (Yin & Yang, 2014). Selain itu kombinasinya dengan logika fuzzy dapat digunakan untuk mengefisienkan penggunaan energy transmisi data pada *wireless sensor network* (Razzaq & Shin, 2019).

Dari hasil penjabaran penelitian-penelitian terdahulu, diketahui bahwa algoritme Dijkstra telah dan dapat diterapkan untuk penyelesaian banyak masalah. Maka dari itu, dari tingkat penggunaannya, algoritme ini dianggap cocok digunakan sebagai

penentu jarak terpendek bagi wisatawan dalam mencapai tujuan wisata di Kota Toboali.

## 2. METODOLOGI PENELITIAN

Langkah-langkah pelaksanaan penelitian ini ditampilkan sebagaimana Gambar 1.



Gambar 1. Tahapan Penelitian

### 2.1. Tahap Pengumpulan Data

Pada tahap ini dilakukan pengumpulan data seperti studi literatur dari buku dan penelitian terdahulu. Selain itu juga dikumpulkan data-data yang dibutuhkan mengenai studi kasus tujuan wisata di Kota Toboali.

### 2.2. Tahap Analisis Algoritme Dijkstra

Tahapan ini mempelajari dan melakukan ujicoba perhitungan manual dan melakukan penerapannya pada sistem berbasis *web* yang dibangun.

### 2.3. Tahap Perancangan Sistem

Pada tahapan ini perancangan sistem dilakukan. Rancangan yang dibuat menggunakan *tool* diagram UML. Hal ini bertujuan agar sistem yang dibangun sesuai dengan kebutuhan dan dapat menjawab permasalahan yang ada.

### 2.4. Tahap Implementasi dan Pengujian Sistem

Hasil perancangan diterapkan dan dibangun sesuai dengan perancangan. Setelah sistem selesai dibangun, langkah selanjutnya adalah melakukan pengujian sistem. Pengujian dilakukan untuk mengetahui sejauh mana kinerja dan dampak dari sistem yang telah dibangun.

## 3. HASIL DAN PEMBAHASAN

### 3.1. Algoritme Dijkstra

Algoritme Dijkstra merupakan algoritme rakus (*greedy*) yang dapat digunakan untuk menyelesaikan penentuan jarak terpendek. Penentuan dapat diterapkan pada *graph* berarah (*directed*) dengan ketentuan bobot/nilai sisinya tidak bernilai negatif (bobot  $\geq 0$ ) (Raja et al., 2015). Algoritme ini menentukan rute dengan jarak terpendek antara tiap titik/*node*. Juga dapat digunakan untuk menentukan total jarak dari suatu titik ke titik tujuan tertentu (Wahyuningsih & Syahreza, 2018). Cara kerja algoritme ini adalah menentukan titik awalnya ( $x$ ), kemudian menentukan jarak terpendek ke titik

selanjutnya yang terdekat dengan titik awal ( $x$ ). Demikian seterusnya sehingga didapat jarak terpendek dari  $x$  ke titik tujuan ( $z$ ) dari berbagai kemungkinan jalur yang tersedia diantaranya (Rifanti, 2017).

Rumus umum dari algoritme ini ditunjukkan pada Persamaan 1.

$$G = (V, E) \quad (1)$$

Dengan:

$G$  : *Graph*

$V$  : *Vertices* (titik/node)

$E$  : *Edge* (jarak/sisi)

Alur proses algoritme Dijkstra adalah sebagai berikut (Hidayat, 2016):

- Tentukan bobot antar titik.
- Tentukan nilai titik awal = 0. Sedangkan titik lainnya “tak hingga”
- Ubah status titik awal menjadi “*node* awal”, sedangkan titik lainnya diberi status “belum terjamah”.
- Dari *node* awal, perhatikan titik-titik terdekat yang belum dilalui dan hitung jaraknya. Lakukan perbandingan total bobot yang didapat. Apabila didapat hasil lebih kecil dari jarak antar titik lainnya, maka hapus data lama dan simpan data baru.
- Setelah poin  $d$ , ubah status titik yang telah dihitung menjadi “terjamah”. Titik-titik dengan status ini tidak akan diperiksa kembali. Adapun jarak yang disimpan adalah perhitungan jarak terakhir dan memiliki nilai bobot paling kecil (rendah).
- Ubah titik “belum terjamah” dengan bobot terkecil menjadi “*node* awal” selanjutnya.
- Kembali ke langkah  $d$  dan lakukan langkah ini sampai seluruh titik didapat bobotnya dan tiba pada titik tujuan.

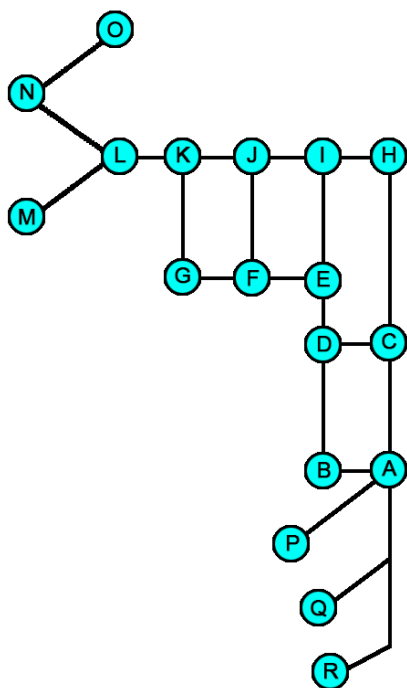
### 3.2. Penentuan Graph

*Graph* yang digunakan adalah *graph* tujuan wisata yang berada di dalam Kota Toboali, yakni Batu Belimbing, Pantai Batu Perahu, Pantai Batu Kapur, Benteng Toboali, dan Pantai Nek Aji. Bentuk *graph* ditampilkan pada Gambar 2.

Penjelasan titik–titik pada Gambar 2 adalah sebagai berikut:

- Titik A, B, C, D, E, F, G, H, I, J, K, dan L adalah titik-titik persimpangan jalan dari kemungkinan rute yang dapat dilewati untuk dapat sampai pada tujuan wisata.
- Titik O : Wisata Batu Belimbing
- Titik N : Pantai Batu Perahu
- Titik M : Pantai Batu Kapur
- Titik P : Benteng Toboali
- Titik Q : Pantai Nek Aji

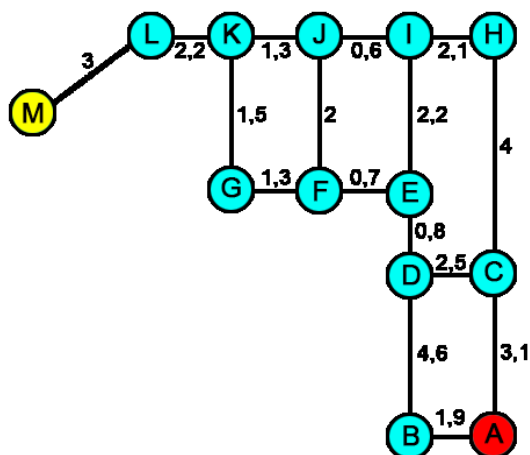
g. Titik R : Wisata Tanjung Batu



Gambar 2. Graph Rute Wisata Kota Toboali

Gambar 2 merupakan bentuk simulasi visual dari jalur-jalur yang tersedia menuju tujuan wisata di Kota Toboali. Simulasi dibuat seperti Gambar 2 dengan tujuan agar lebih mudah dalam melakukan pemahaman jarak.

Untuk melakukan penentuan jarak terpendek dengan algoritme Dijkstra, kita melakukannya dengan mengikuti alur proses seperti yang telah dijabarkan sebelumnya. Dari Gambar 2, akan diambil kasus perhitungan dari titik A (Titik awal) menuju titik M (Pantai Batu Kapur) dari berbagai kemungkinan rute yang ada dan ditampilkan pada Gambar 3.



Gambar 3. Graph untuk contoh perhitungan

Gambar 3 menunjukkan jalur yang digunakan untuk perhitungan telah dilengkapi dengan nilai jarak antar titik. Penentuan nilai dari jarak

sesungguhnya antar titik dibagi dengan 100. Hal ini dilakukan agar lebih mudah dalam melakukan perhitungan.

Sesuai dengan alur proses pada subbab 3.1, maka proses perhitungannya adalah:

- Tentukan bobot antar titik. Bobot antar titik didapat dari jarak sesungguhnya dibagi 100. Hal ini dilakukan untuk memudahkan perhitungan.
- Tentukan nilai titik awal = 0. Beri nilai titik lainnya "tak hingga". Titik awal pada titik A dan titik tujuan adalah M.
- Ubah status titik awal menjadi "node awal", jadi titik A kita beri status "node awal", sedangkan titik lainnya diberi status "belum terjamah".
- Dari node awal, perhatikan titik-titik terdekat yang belum dilalui dan hitung jaraknya. Dari titik A ada kemungkinan ke titik B dan C. Lakukan perbandingan total bobot yang didapat ( $B=1,9$  :  $C=3,1$ ). Karena nilai bobot AB lebih kecil dari AC, maka kita pilih bobot AB, yakni 1,9. Kemudian dilanjutkan ke D (4,6) menjadi ABD sehingga total bobot menjadi 6,5. Karena algoritme ini adalah algoritme *greedy*, maka kita harus melakukan perhitungan AC juga. Dari titik C ada kemungkinan titik D dan H. Kita pilih titik D dengan bobot terendah ( $D=2,5$  :  $H=4$ ). Jadi total bobot  $ACD = 3,1+2,5 = 5,6$ . Karena nilai ACD lebih kecil daripada ABD, maka kita pilih ACD. Hapus data ABD, karena berbobot besar.
- Setelah poin d, ubah status titik yang telah dihitung menjadi "terjamah". Ini artinya titik B dan D menjadi titik "terjamah" dan tidak akan diperiksa kembali.
- Ubah titik D menjadi "node awal". Dan lakukan perhitungan dari poin (d) dari titik D sampai dengan tercapai titik tujuan M.

Hasil dari perhitungan dijkstra terhadap Gambar 2 dijabarkan pada Tabel 1.

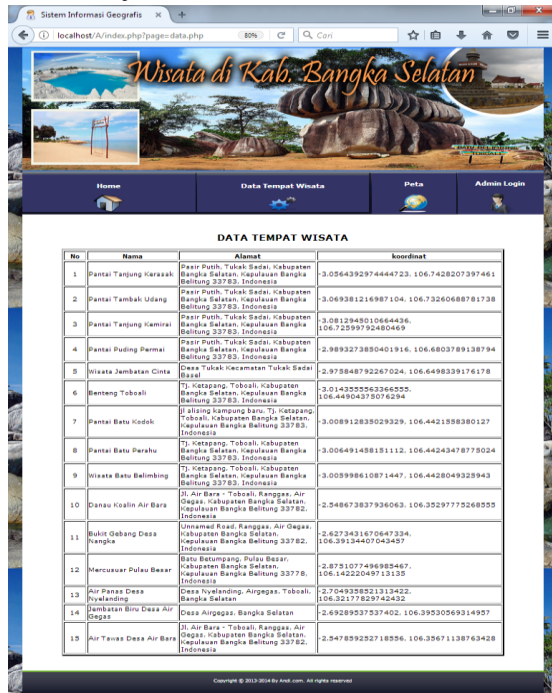
Titik	Status	Bobot	Rute
A	Terjamah	0	A
(awal)			
B	Terjamah	1,9	AB
D	Terjamah	6,5	ABD
C	Terjamah	3,1	AC
D	Terjamah	5,6	ACD
H	Terjamah	7,1	ACH
E	Terjamah	6,4	ACDE
F	Terjamah	7,1	ACDEF
I	Terjamah	8,6	ACDEI
G	Terjamah	8,4	ACDEFG
K	Terjamah	9,9	ACDEFGK
J	Terjamah	10,4	ACDEFGJ
L	Terjamah	12,1	ACDEFGKL
M	Terjamah	15,1	ACDEFGKLM

Dari Tabel 1 didapat hasil bahwa rute terpendek titik A ke titik M adalah melalui jalur  $A \rightarrow C \rightarrow D \rightarrow E \rightarrow F \rightarrow G \rightarrow K \rightarrow L \rightarrow M$  dengan total bobot jalur adalah 15,1 (1.510 m).



### 3.3. Tampilan Antarmuka

#### 1. Data Tujuan Wisata

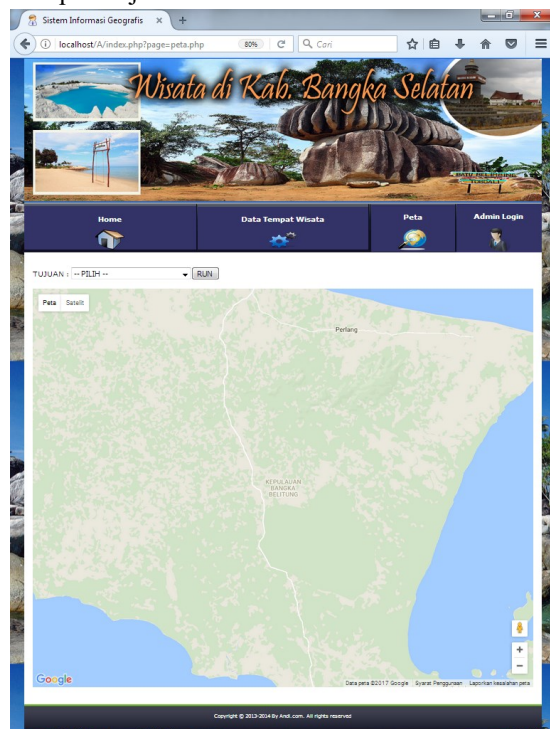


No	Nama	Alamat	koordinat
1	Pantai Tanjung Kerak	Pesir Putih, Tukak Sada, Kabupaten Bangka Selatan, Kepulauan Bangka Belitung 33762, Indonesia	-3.0564392974444723, 106.7428207397481
2	Pantai Tembak Udag	Pesir Putih, Tukak Sada, Kabupaten Bangka Selatan, Kepulauan Bangka Belitung 33762, Indonesia	-3.069381216987104, 106.73260688781738
3	Pantai Tanjung Kemirai	Pesir Putih, Tukak Sada, Kabupaten Bangka Selatan, Kepulauan Bangka Belitung 33762, Indonesia	-3.0812948010664436, 106.72599792480469
4	Pantai Puding Pama	Pesir Putih, Tukak Sada, Kabupaten Bangka Selatan, Kepulauan Bangka Belitung 33762, Indonesia	-2.9893273850401916, 106.4803789138794
5	Wisata Jembatan Cinta	Casa Tukak Nacemmen, Tukak Sada, Kabupaten Bangka Selatan, Kepulauan Bangka Belitung 33762, Indonesia	-2.97848792267024, 106.6498339176178
6	Bentang Toboali	Tj. Ketapang, Toboali, Kabupaten Bangka Selatan, Kepulauan Bangka Belitung 33762, Indonesia	-3.0143550563366555, 106.449504379076284
7	Pantai Batu Kodok	3 siliang Lampung Baru, Tj. Ketapang, Toboali, Kabupaten Bangka Selatan, Kepulauan Bangka Belitung 33762, Indonesia	-3.008912835029329, 106.4421598380127
8	Pantai Batu Perahu	Tj. Ketapang, Toboali, Kabupaten Bangka Selatan, Kepulauan Bangka Belitung 33762, Indonesia	-3.006491486181112, 106.44243478778024
9	Wisata Batu Balinging	Tj. Ketapang, Toboali, Kabupaten Bangka Selatan, Kepulauan Bangka Belitung 33762, Indonesia	-3.008998610871447, 106.4428049328943
10	Danau Kialin Air Bara	Jl. Air Bara - Toboali, Rangas, Air Gagap, Kabupaten Bangka Selatan, Kepulauan Bangka Belitung 33762, Indonesia	-2.548673837936063, 106.35297775268505
11	Rukin Gading Desa Rangka	Unusmed Road, Rangas, Air Gagap, Kabupaten Bangka Selatan, Kepulauan Bangka Belitung 33762, Indonesia	-2.6273431670647234, 106.39134407043457
12	Mercusuar Pulau Besar	Batu Belatung, Pulau Besar, Kabupaten Bangka Selatan, Kepulauan Bangka Belitung 33762, Indonesia	-2.8781077496985467, 106.14222049713135
13	Air Panas Desa Nyalending	Casa Nyalending, Airgagas, Toboali, Bangka Selatan	-2.7049358521313422, 106.32177829742432
14	Kampayan Biru Desa Air Gagap	Casa Airgagas, Bangka Selatan	-2.69289537537402, 106.39530569314957
15	Air Tawar Desa Air Bara	Jl. Air Bara - Toboali, Rangas, Air Gagap, Kabupaten Bangka Selatan, Kepulauan Bangka Belitung 33762, Indonesia	-2.547859232718956, 106.3567118763428

Gambar 4. Tampilan Halaman Data Tujuan Wisata

Gambar 4 menampilkan halaman data tujuan wisata. Pengaksesan halaman ini memungkinkan pengguna untuk memperoleh informasi mengenai data tujuan wisata yang ada di Kota Toboali dan seluruh Kabupaten Bangka Selatan, lengkap dengan alamat dan koordinat lokasinya.

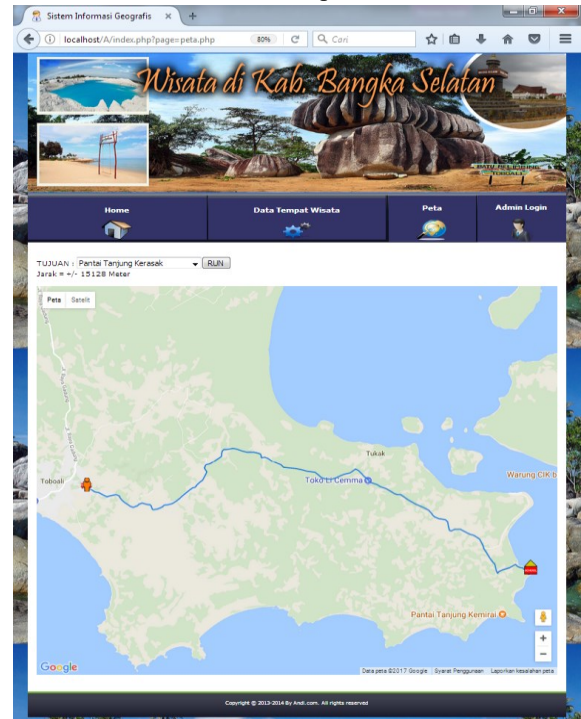
#### 2. Input Tujuan Wisata



Gambar 5. Halaman Input Titik Awal dan Tujuan

Gambar 5 merupakan halaman bagi pengguna untuk memasukkan lokasi awal mereka, selain itu pengguna dapat memilih titik tujuan wisata mereka sesuai dengan keinginan pengguna berdasar pilihan wisata yang tersedia.

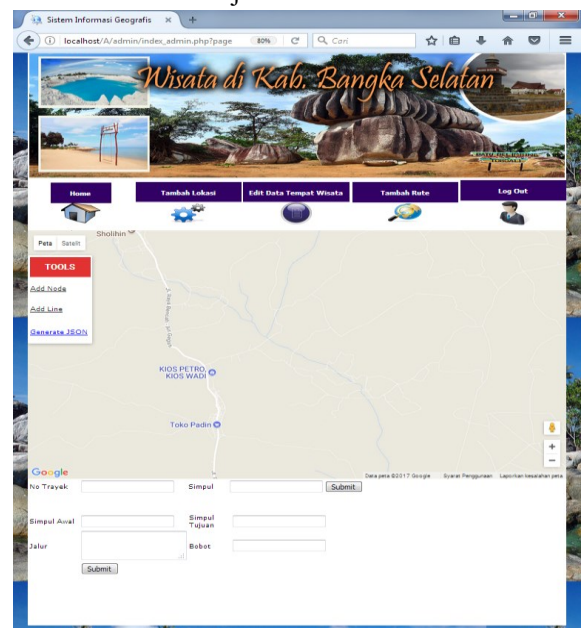
#### 3. Hasil Pencarian Rute Terpendek



Gambar 6. Hasil Pencarian Rute Terpendek

Gambar 6 menampilkan hasil pencarian rute terpendek berdasarkan titik awal dan titik tujuan yang telah ditentukan sebelumnya.

#### 4. Tambah Rute Tujuan



Gambar 7. Halaman Input Rute Tujuan Wisata Baru

Gambar 7 menampilkan halaman yang digunakan oleh Admin untuk menambahkan rute tujuan wisata baru yang belum tersimpan pada *database*. Admin dapat memasukkan kemungkinan simpul atau titik yang dilalui untuk mencapai tujuan wisata. Selain itu dimasukkan juga jalur atau rute dan bobot dari tiap titik.

### 3.4. Pengujian

Pada tahapan ini dilakukan pengujian terhadap sistem. Tujuan dari pengujian adalah untuk mengetahui kinerja dari sistem yang dibangun. Pengujian dilakukan dengan menggunakan metode *Blackbox* yang berfungsi untuk mengetahui apakah fungsional sistem bekerja atau tidak. Hasil dari pengujian dijabarkan pada Tabel 2.

Tabel 2. Pengujian *Blackbox*

Pengujian	Hasil	Validasi
Menu Home	Aplikasi diarahkan ke halaman utama	Berhasil
Menu Data	Tampil halaman data informasi	Berhasil
Tempat Wisata	tujuan wisata berupa tabel	
Menu Peta	Tampil halaman input tujuan wisata dengan pilihan wisata yang dapat dipilih	Berhasil
Input Tujuan Wisata	Tujuan wisata terpilih, muncul perkiraan jarak dari titik awal menuju tujuan, dan muncul rute terpendek yang dapat dilalui	Berhasil
Penambahan titik tujuan wisata	Muncul form tambah lokasi untuk input data lokasi tujuan wisata baru berisi koordinat titik	Berhasil
Penambahan rute ke tujuan wisata	Muncul form tambah rute berisi simpul awal, simpul tujuan, jalur, dan bobot	Berhasil

## 4. KESIMPULAN

Sistem Informasi Geografis Pariwisata Kabupaten Bangka Selatan ini telah menerapkan fungsi pencarian rute terpendek menggunakan algoritme Dijkstra. Dari hasil penerapan, sistem mampu menentukan jarak terpendek dari titik awal pengguna menuju titik tujuan wisata pengguna. Selain itu dari hasil perhitungan manual berdasarkan Gambar 3 pada tujuan wisata berfokus di Kota Toboali dibuktikan pula bahwa Algoritme Dijkstra dapat menentukan bobot terendah untuk mencapai jarak terpendek menuju titik tujuan. Berdasarkan perhitungan manual sebagaimana Tabel 1 didapat hasil terpendek dari titik A menuju titik M yakni melalui rute  $A \rightarrow C \rightarrow D \rightarrow E \rightarrow F \rightarrow G \rightarrow K \rightarrow L \rightarrow M$  dengan total bobot jalur yang dilalui adalah 15,1 (1.510 m).

Untuk penelitian selanjutnya disarankan untuk melakukan penerapan pada perangkat *mobile*. Juga disarankan untuk melakukan pengujian tingkat lanjut terhadap kinerja algoritme, dan pengujian beta terhadap pengalaman pengguna sistem.

## DAFTAR PUSTAKA

- ARDHIANSYAH, M., PUTRA, M., GINARDI, R. V. H., & MUNIF, A. (2016). Sistem Navigasi Indoor Menggunakan Bi- Directional Dijkstra Search Berbasis Integrasi dengan Smartphone untuk Studi Kasus pada Gedung Bertingkat. *Jurnal Teknik ITS*, 5(2), A547–A551.
- BOYZYIGIT, A., ALANKUS, G., & NASIBOGLU, E. (2017). Public Transport Route Planning: Modified Dijkstra Algorithm. In *International Conference on Computer Science and Engineering* (pp. 502–505). Antalya: IEEE. <https://doi.org/10.1109/UBMK.2017.8093444>
- GINARDI, R. V. H., & MUNIF, A. (2016). Penggunaan QR Code untuk Menunjukkan Posisi dan Implementasi Algoritme Dijkstra dalam Pencarian Rute Terpendek pada Navigasi dalam Ruangan Berbasis Sistem Operasi Android. *Jurnal Teknik ITS*, 5(2), 604–607.
- GOO, B., JUNG, S., & HUR, J. (2016). Development of a Sequential Restoration Strategy Based on the Enhanced Dijkstra Algorithm for Korean Power Systems. *Applied Science*, 6(435), 1–11. <https://doi.org/10.3390/app6120435>
- HIDAYAT, H. T. (2016). SISTEM PENUNJANG KEPUTUSAN PENCARIAN JARAK TERPENDEK MENUJU RUMAH SAKIT DAN PUSKESMAS DENGAN METODE DIJKSTRA. *Jurnal Techsi*, 8(1), 95–124. <https://doi.org/10.29103/techsi.v8i1.119>
- HOSSAIN, S., DAS, N., PATWARY, M. K. H., & HASSAN, M. AL. (2018). Finding the Nearest Blood Donors using Dijkstra Algorithm. *Journal of Information System (Sisforma)*, 5(2), 40–44. <https://doi.org/10.24167/Sisforma.v5i2.1709>
- HOU, D., & ZHANG, W. (2017). Multi - Warehouse Location of Logistics Based on Dijkstra and Divide-and- Conquer Algorithm. In *International Symposium on Computational Intelligence and Design* (pp. 442–447). Hangzhou: IEEE. <https://doi.org/10.1109/ISCID.2017.204>
- KURNIAWAN, M. R., NURHAYATI, O. D., & MARTONO, K. T. (2015). SISTEM INFORMASI GEOGRAFIS PENCARIAN LOKASI AGEN BUS DAN TRAVEL TERDEKAT DI KOTA SEMARANG BERBASIS MOBILE DENGAN METODE DIJKSTRA. *Jurnal Teknologi Dan Sistem Komputer*, 3(2), 302–310.
- KUSUMA, E., & AGUNG, H. (2019). Aplikasi Perhitungan Dan Visualisasi Jarak Terpendek Berdasarkan Data Coordinate Dengan Algoritme Dijkstra Dalam Kasus Pengantaran Barang Di Kawasan Jabodetabek. *Jurnal*

- Sisfokom*, 08(1), 14–23.
- MIEEE, A. M. A.-, AHMED, S. H., & AHMED, O. H. (2017). Dijkstra Algorithm Applied : Design and Implementation of a framework to find nearest Hotels and Booking Systems in Iraqi. In *International Conference on Current Research in Computer Science and Information Technology* (pp. 126–132). Slemani: IEEE. <https://doi.org/10.1109/CRCST.2017.7965546>
- MOO, F., HERNÁNDEZ, R., UC, V., & MADERA, F. (2016). Web Service Composition Using The Bidirectional Dijkstra Algorithm. *IEEE LATIN AMERICA TRANSACTIONS*, 14(5), 2522–2528.
- NONADP, N. (2017). 11 Program Unggulan Menjadi Prioritas RPJMD Babel 2017-2022. Retrieved March 17, 2019, from <https://babelprov.go.id/content/11-program-unggulan-menjadi-prioritas-rpjmd-babel-2017-2022>
- OJEKUDO, N. A., & AKPAN, N. P. (2017). Anapplication of Dijkstra ' s Algorithm to shortest route problem . *IOSR Journal of Mathematics*, 13(3), 20–32. <https://doi.org/10.9790/5728-1303012032>
- PERDA BASEL, B. S. PERATURAN DAERAH KABUPATEN BANGKA SELATAN NOMOR 11 TAHUN 2016 TENTANG PENYELENGGARAAN KEPARIWISATAAN (2016). Indonesia.
- PRASAD, A., KRISHNAMURTHY, S. K., & KIM, Y. (2018). Acceleration of Dijkstra ' s Algorithm on Multi-core Processors. In *International Conference on Electronics, Information, and Communication* (pp. 6–10). Honolulu: IEEE. <https://doi.org/10.23919/ELINFOCOM.2018.8330701>
- PRIANTO, C., & KUSNADI, M. (2018). Penerapan Algoritme Dijkstra Untuk Menentukan Rute Terbaik Pada Mobile E-Parking Berbasis Sistem Informasi Geografis. *Jurnal Informatika: Jurnal Pengembangan IT*, 03(03), 329–335. <https://doi.org/10.30591/jpit.v3i3.941>
- QING, G., ZHENG, Z., & YUE, X. (2017). Path-planning of Automated Guided Vehicle based on Improved Dijkstra Algorithm. In *Chinese Control And Decision Conference* (pp. 7138–7143). Chongqing: IEEE. <https://doi.org/10.1109/CCDC.2017.7978471>
- RAJA, B. H. P. M., N, A. B. P., & IRWANSYAH, M. A. (2015). Rancang Bangun Sistem Informasi Geografis Berbasis Web Fasilitas Pelayanan Kesehatan di Kota Pontianak. *Jurnal Edukasi Dan Penelitian Informatika*, 1(2), 64–71.
- RAKYAT POS, R. (2017). Pengembangan dan Strategi Wisata Bahari Bangka Selatan. Retrieved March 8, 2019, from <http://www.rakyatpos.com/pengembangan-dan-strategi-wisata-bahari-bangka-selatan.html/>
- RAZZAQ, M., & SHIN, S. (2019). Fuzzy-Logic Dijkstra-Based Energy-Efficient Algorithm for Data Transmission in WSNs. *Sensors*, 19(1040), 1–22. <https://doi.org/10.3390/s19051040>
- RIFANTI, U. M. (2017). PEMILIHAN RUTE TERBAIK MENGGUNAKAN ALGORITME DIJKSTRA UNTUK MENGURANGI KEMACETAN LALU LINTAS DI PURWOKERTO. *Jurnal Matematika Dan Pendidikan Matematika*, 2(2), 90–99.
- RISALD, R., MIRINO, A. E., & SUYOTO, S. (2017). Best Routes Selection Using Dijkstra Algorithm and Floyd-Warshall Algorithm. In *International Conference on Information & Communication Technology and System* (pp. 155–158). Surabaya: IEEE. <https://doi.org/10.1109/ICTS.2017.8265662>
- VESOVI, M., SMILJANI, A., & KOSTI, D. (2016). Performance of Shortest Path Algorithm Based on Parallel Vertex Traversal. *SERBIAN JOURNAL OF ELECTRICAL ENGINEERING*, 13(1), 31–43. <https://doi.org/10.2298/SJEE1601031V>
- WAHYUNINGSIH, D., & SYAHREZA, E. (2018). Shortest Path Search Futsal Field Location With Dijkstra Algorithm. *IJCCS*, 12(2), 161–170. <https://doi.org/10.22146/ijccs.34513>
- WANG, H., ZHANG, F., & CUI, P. (2017). A Parking lot induction method based on Dijkstra algorithm. In *Chinese Automation Congress* (pp. 5247–5251). Jinan: IEEE. <https://doi.org/10.1109/CAC.2017.8243712>
- WU, M., WU, B., & SONG, H. (2017). Application of Java - Based Optimization Dijkstra Algorithm in Parking Lot Berth Guidance. In *International Conference on Software Engineering and Service Science* (pp. 237–240). Beijing: IEEE. <https://doi.org/10.1109/ICSESS.2017.8342904>
- YIN, X., & YANG, J. (2014). Shortest Paths Based Web Service Selection in Internet of Things. *Journal of Science*, 2014, 1–10. <https://doi.org/http://dx.doi.org/10.1155/2014/958350>
- ZHANG, JIN-DONG, FENG, Y., SHI, F., WANG, G., LI, R., & JIA, X. (2016). Vehicle routing in urban areas based on the Oil Consumption Weight-Dijkstra algorithm. *The Institution of Engineering and Technology Journal*, 10(7), 495–502. <https://doi.org/10.1049/iet->

its.2015.0168

- ZHANG, JING, YU, J., QU, X., & WU, Y. (2017). Path Planning for Carrier Aircraft Based on Geometry and Dijkstra ' s Algorithm. In *International Conference on Control Science and Systems Engineering* (pp. 115–119). Beijing: IEEE.  
<https://doi.org/10.1109/CCSSE.2017.8087906>
- ZHANG, Y., SU, Y., YANG, J., PONCE, I. J., & KONG, H. (2018). When Dijkstra meets vanishing point : a stereo vision approach for road detection. *JOURNAL OF LATEX CLASS FILES*, 14(8), 12.  
<https://doi.org/10.1109/TIP.2018.2792910>
- ZHU, D., DU, H., SUN, Y., & CAO, N. (2018). Research on Path Planning Model Based on Short-Term Traffic Flow Prediction in Intelligent Transportation System. *Sensors*, 18, 1–15. <https://doi.org/10.3390/s18124275>
- ZULFIQAR, L. O. M., ISNANTO, R. R., & NURHAYATI, O. D. (2018). Optimal Distribution Route Planning based on Collaboration of Dijkstra and Sweep Algorithm. In *International Conference on Information Technology and Electrical Engineering* (pp. 371–375). Kuta: IEEE.